I. AMENDMENTS TO THE CLAIMS

Please find below a listing of claims that will replace all prior versions, and listings, of claims in the application:

1. – 5. (*cancelled*)

6. (*currently amended*) A method of executing a set of at least one incomplete task, comprising:
   (a) selecting an incomplete task from the set on the basis of an expected duration until completion for that task;
   (b) resetting an execution timer having an expiry condition;
   (c) advancing execution of the selected task until the earlier of (i) completion of the selected task or (ii) expiry of the execution timer; and
   (d) upon expiry of the execution timer prior to completion of the selected task, suspending execution of the selected task.

7. – 10. (*cancelled*)

11. (*previously presented*) A method as defined in claim 6, wherein advancing execution of the selected task includes beginning the selected task if the selected task has not been previously suspended.

12. (*cancelled*)

13. (*original*) A method as defined in claim 11, wherein advancing execution of the selected task includes resuming the selected task if the selected task has been previously suspended.

14. (*original*) A method as defined in claim 13, wherein suspending the selected task includes saving a context associated with the selected task.

15. (*original*) A method as defined in claim 14, wherein resuming the selected task includes retrieving the previously saved context associated with the selected task.

16. (*original*) A method as defined in claim 15, wherein the context associated with the selected task includes variables local to the selected task.

17. (*original*) A method as defined in claim 15, wherein the context associated with the selected task includes a state of the selected task.

18. (*original*) A method as defined in claim 15, wherein the context associated with the selected task includes a state of a central processing unit (CPU).

19. (*previously presented*) A method as defined in claim 6, wherein the expiry condition of the execution timer is a pre-determined number of clock cycles.

20. (*previously presented*) A method as defined in claim 6, wherein the expiry condition of the execution timer is a pre-determined period of time.

21. (*previously presented*) A method as defined in claim 6, wherein the expiry condition of the execution timer is a pre-determined percentage of completeness of the selected task.

22. (*currently amended*) A method of executing a set of incomplete tasks, comprising:

(a) removing an existing incomplete task from the set when a newer version of the existing incomplete task is added to the set;

(b) executing the remainder of the set of incomplete tasks[[.]];

(c) <u>wherein said removing is effected without completing said existing incomplete task.</u>

23. – 40. (*cancelled*)

41. (*currently amended*) A method [[~~as defined in claim 8,~~]] of executing a set of at least one incomplete task, comprising:

(a) selecting an incomplete task from the set on the basis of a number of times that the task has been previously suspended;

(b) resetting an execution timer having an expiry condition, wherein the expiry condition of the execution timer is a pre-determined percentage of completeness of the selected task[[~~,~~]];

(c) advancing execution of the selected task until the earlier of (i) completion of the selected task or (ii) expiry of the execution timer; and

(d) upon expiry of the execution timer prior to completion of the selected task, suspending execution of the selected task.